

# “QR Code Generator”

## Personal Details

Name	Shubham Goyal
Email	22shubh22@gmail.com
IRC Nick	shubhamg
Country	India
TimeZone	UTC +5:30

---

## Project Abstraction

### Current status

There is no feature in LO to make a QR Code. If one needs to add a QR code to their doc then they have to use some other techniques to generate and add a QR code.

The Project idea aims at integrating a Library to generate QR code within LibreOffice itself and add options to use the QR code in LO applications.

- QR Code will be generated for text or URL.
  - QR Code can be generated even for special(UTF) characters.
  - The generated image will be in SVG Format. Better Rendering Quality, Scalable :)
  - QR Code will be generated as an Image so it is handled easily like an Image.
- 

## Why the Project?

Generating QR code is required as a feature in core.

### Some justifications

- [tdf#87195](#) - mail merge: add image from url field
- See how Microsoft Office as QR code generated in core [here](#).

### Bugzilla Tickets

- [tdf#62168](#) Print and Display hyperlinks as QR option.

---

## Implementation Details

### Library Details

Library to be used - <https://github.com/nayuki/QR-Code-generator>

Live Demo for Library - <https://www.nayuki.io/page/qr-code-generator-library>

API to be used ( those to be included in the project )

```
const QrCode qr = QrCode::encodeText(text, errCorLv1);
```

^ to generate the qr code

parameters are Text and Error Level control

- Text - String whose QR code is to be generated.

To deal with problem of interchange of std:: string and OUString, the following hyperlink can help

<https://forum.openoffice.org/en/forum/viewtopic.php?f=44&t=54978#>

- Error Level Control

Defines the error correction code (ECC) which determines the degree of data redundancy. The more data redundancy exists, the more data can be restored if a QR code is damaged (i.e. scratches on a QR code sticker or something like that).

Possible values:

- L (low, ~7% destroyed data may be corrected)
- M (middle, ~15% destroyed data may be corrected)
- Q (quality, ~25% destroyed data may be corrected)
- H (high, ~30% destroyed data may be corrected)

```
qr.toSvgString(4);
```

^ API to convert QR into an SVG Image , provided by library

Take border as a input.

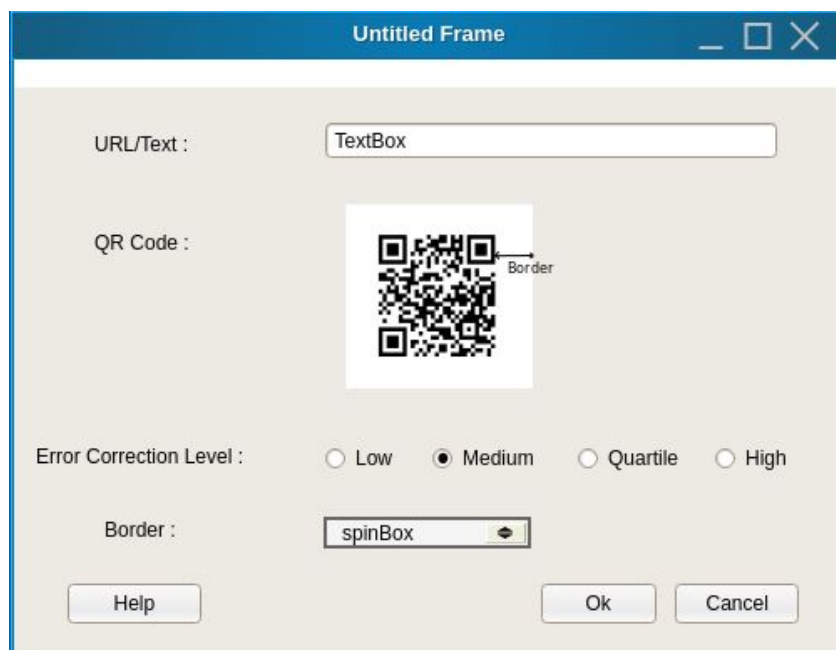
---

## Dialog Box

A Dialog Box will act as a interface to interact with QR code generating Library.

The following inputs are needed.

Input	About	UI Object
URL/Text	The String whose URL is to made	GtkEntry
Error Correction Level	Level of Error Correction In QR Code Generation	4 Radio Buttons
Border	Integer - Whitespace with around the QR code	SpinButton



## **WorkFlow**

Input URL and other requirements are taken from Dialog Box.

The input from the Dialog Box is then processed with the Library to form SVG Image.

---

## **Location of Processing File**

After discussing from Samuel, it was found that the above workflow can be implemented the same as the implementation of "SignatureLineDialog".

At location cui/source/dialogs new files have to be generated.

The location is chosen because feature from here will be accessible to all LO applications.

Look at the initial demo changes made for this project at

<https://gerrit.libreoffice.org/#/c/70002/>

---

## **Generating QR Code Image in Application**

1. The QR code dialog should be accessible at Insert menu in all the LO applications.

2. Fill QR code dialog Box and click OK. Then the QR Code Image should be inserted at the at the cursor position.

---